Technical Note: Global LSH Gating as a Mechanism for Improved Weight Utilization in Sparse Neural Networks

Executive Summary

This note proposes that **global locality-sensitive hash (LSH) gating**, determined entirely from the network input, may improve learning stability and learning speed compared with conventional local gating mechanisms such as ReLU.

The central hypothesis is that:

> A fixed, global, locality-preserving gating function may distribute training updates more uniformly across network parameters, thereby improving parameter utilization, reducing pathway collapse, and producing smoother input-dependent effective mappings.

The proposal is speculative but experimentally straightforward.

Background

Conventional Local Gating

Modern sparse neural computation often uses local gates:

x
↓
Layer 1 + ReLU
↓
Layer 2 + ReLU
↓
Layer 3 + ReLU

Each gate decision depends on intermediate activations.

Consequences:

- routing decisions are recursive;
- later routing depends on earlier routing;
- pathways can reinforce themselves;
- some neurons may become permanently inactive ("dead neurons");
- parameter usage may become highly non-uniform.

Effective Matrix View of Neural Networks

A feedforward network with input-dependent gates can be viewed as implementing:

$$y = M(x)x$$

where:

- x is the input;
- M(x) is an input-dependent effective matrix.

Although the network contains many layers and nonlinearities, for any fixed input the entire network collapses to a single linear mapping.

Different inputs generally produce different effective matrices.

Thus learning may be viewed as constructing a family of matrices:

$$\{M(x)\}$$

indexed by the input space.

This viewpoint suggests that the principal role of gating is to select which parameters contribute to M(x).

Global Gating Proposal

Instead of recursively computing gate decisions layer-by-layer:

input
→ layer 1 gates

→ layer 2 gates
→ layer 3 gates

compute all gates once from the input:

input
→ global routing sketch
→ all network gates

No layer dependencies exist.

All routing decisions are fixed before any hidden-layer computation begins.

Locality-Sensitive Hashing (LSH)

LSH refers to a family of methods satisfying:

> Similar inputs collide with high probability.

A simple example:

1. Generate random projection vectors:

   $r_1, r_2, ..., r_n$

2. Compute:

   $b_i = \text{sign}(r_i \cdot x)$

3. Concatenate the bits:

   101011001101...

This binary code forms an approximate locality-preserving hash.

Nearby inputs tend to produce nearby hashes.

Important Clarification

The term "random projection" is potentially misleading.

After initialization:

- projections are fixed;
- routing is deterministic;
- nearby inputs remain nearby;
- substantial geometric structure is preserved.

Thus the mechanism is not random during operation.

The random initialization merely defines a fixed partitioning of input space.

Central Hypothesis

Global LSH gating may improve training because it gives network parameters a more equitable opportunity to participate in the effective matrix:

$$M(x)$$

for different inputs.

Conventional local gating may exhibit a "rich-get-richer" effect:

frequently active pathways
↓
receive larger updates
↓
become even more dominant

leading to:

- dead neurons;
- pathway monopolization;
- uneven parameter utilization.

Global LSH gating may instead yield:

input distribution
↓
LSH routing
↓
approximately uniform pathway selection

with parameter activation frequencies satisfying:

$P(w_i \text{ active}) \approx P(w_j \text{ active})$

for many weights.

Potential Advantages

Improved parameter utilization

More parameters participate in learning.

Reduced dead neurons

Few parameters become permanently excluded.

Greater training stability

Routing no longer co-adapts with weights.

Reduced optimization feedback loops

Routing statistics become independent of weight evolution.

Smooth functional variation

Nearby inputs produce nearby hashes.

Therefore:

nearby inputs
↓

nearby gates
↓
similar effective matrices
↓
smooth function approximation

Increased model capacity usage

The entire parameter budget may be used more effectively.

Possible Failure Modes

Insufficient task adaptation

Fixed routing cannot specialize.

Excessive randomness

Poorly designed hashes may partition the data unfavorably.

Loss of hierarchical abstraction

Layerwise routing may exploit progressively learned representations.

Hash imbalance

Input distributions may still activate some pathways more frequently than others.

Suggested Experiments

Compare standard ReLU networks against identical architectures using global LSH gating.

Measure:

Learning performance

- training loss;
- validation loss;

- convergence speed.

Stability

- gradient variance;
- sensitivity to initialization;
- run-to-run variance.

Parameter utilization

For every parameter (w_i), record:

- activation frequency;
- update frequency;
- cumulative gradient magnitude.

Compute:

- entropy of parameter usage;
- Gini coefficient;
- fraction of dead parameters.

Effective matrix diversity

Estimate:

M(x)

for many inputs and measure:

- rank;
- overlap between matrices;
- smoothness under input perturbations.

Broader Interpretation

If successful, this approach would suggest that:

> The primary role of gating may not be to learn routing, but rather to ensure stable, locality-preserving, and sufficiently diverse parameter participation in the family of effective mappings {M(x)}.

Keywords and References

Effective network linearization

Keywords:

- linear regions of neural networks
- piecewise linear neural networks
- neural network linearization
- input-dependent effective matrix

Locality-sensitive hashing

Keywords:

- locality sensitive hashing
- SimHash
- random hyperplane LSH
- angular LSH

Random projections

Keywords:

- random projection
- Johnson–Lindenstrauss lemma
- random feature methods

Relevant concept:

Johnson–Lindenstrauss lemma

Sparse computation

Keywords:

- sparse neural networks
- conditional computation
- sparse mixture of experts
- dynamic sparse training

Geometry of data

Keywords:

- manifold hypothesis
- intrinsic dimension
- approximate nearest neighbour graphs

Relevant concept:

Manifold hypothesis

Related architectures

Keywords:

- mixture of experts
- reservoir computing
- echo state networks
- random kitchen sinks
- LSH attention
- conditional computation in neural networks